Data-Driven Insights for Fetal Health Prediction Using Machine Learning

Team Members: Pranav Natrajan, Jayesh Locharla

April 10, 2025

1 Introduction

Fetal health monitoring is a critical aspect of prenatal care, aiming to ensure the well-being of the fetus and prevent adverse birth outcomes. Among the most commonly used clinical tools is Cardiotocography (CTG), which records fetal heart rate (FHR) and uterine contractions to assess fetal distress. However, traditional interpretation of CTG traces often relies on the subjective judgment of clinicians, leading to variability, misdiagnosis, and delays in timely intervention. In high-risk pregnancies, where complications may escalate rapidly, objective and accurate methods of CTG interpretation are essential.

Recent advances in machine learning (ML) offer an opportunity to enhance fetal health assessment through automated classification of CTG signals. ML models can identify subtle patterns and nonlinear relationships in the data that are not apparent to human observers. This capability has the potential to support clinical decision-making, reduce diagnostic errors, and contribute to improved maternal and neonatal outcomes.

This project explores the application of various supervised learning techniques to classify fetal health status based on CTG features. Specifically, we aim to classify fetal health into three categories: Normal, Suspect, and Pathological, using a labeled dataset from the UCI Machine Learning Repository. The dataset contains 2,126 samples with 21 numerical features extracted from CTG signals. These features include baseline FHR, accelerations, decelerations, and several histogram-based descriptors.

Our approach investigates both traditional machine learning algorithms, such as Support Vector Machines (SVM), and deep learning models, particularly Artificial Neural Networks (ANN). We further benchmark these models against XGBoost, a high-performance gradient boosting algorithm known for its robustness and interpretability. To tackle the inherent class imbalance in the dataset, we explore two strategies: Synthetic Minority Over-sampling Technique (SMOTE) and cost-sensitive learning, which penalizes the misclassification of underrepresented classes during training.

In addition to classification performance, model interpretability plays a vital role in medical applications. We incorporate SHAP (SHapley Additive exPlanations) to provide global and local explanations of feature contributions, ensuring transparency and trust in the model's predictions. Moreover, we apply Principal Component Analysis (PCA) for visual exploration of data separability.

The goal of this study is to develop an accurate, interpretable, and clinically applicable ML framework for fetal health classification. Through comparative analysis of multiple models and techniques, we aim to identify the most effective strategies for early and reliable prediction of fetal complications. Our findings are expected to inform future research and serve as a foundation for building real-time diagnostic tools in obstetric care.

2 Literature Review

[1] The study by Deng et al. (2023) introduces LW-FHRNet, a lightweight fetal distress-assisted diagnosis model that utilizes a cross-channel interactive attention mechanism and wavelet packet decomposition to improve FHR signal classification. Unlike traditional deep learning models that demand high computational resources, LW-FHRNet efficiently transforms one-dimensional FHR signals into two-dimensional feature maps for enhanced pattern recognition while maintaining a small parameter size (0.33M). Achieving 95.24% accuracy on the CTU-UHB dataset, this model demonstrates the feasibility of using lightweight architectures for fetal health monitoring. This research is highly relevant to our study, as it highlights the benefits of feature extraction techniques and efficient classification models in detecting fetal distress. By integrating wavelet packet decomposition and exploring model efficiency, our project can enhance SVM and ANN approaches, leading to improved fetal health classification and real-time clinical applications.

[2] The study by Mandala (2024) introduces a LightGBM-based approach for fetal health classification, achieving 98.31% accuracy by leveraging features such as fetal heart rate, uterine contractions, and maternal blood pressure. It highlights the limitations of traditional CTG interpretation and demonstrates the benefits of feature selection, model optimization, and class balancing using SMOTE. This research aligns closely

with our study, which aims to implement SVM and ANN models for fetal health classification. Comparing the performance of LightGBM with SVM and ANN will provide valuable insights into the effectiveness of different classification techniques. Additionally, the study's emphasis on feature importance and preprocessing methods supports our approach in identifying key CTG attributes for better pattern recognition in fetal health assessment. By integrating these findings, our study aims to enhance machine learning-driven fetal health monitoring and improve clinical decision-making.

[4] The study by Sulistianingsih and Martono (2024) explores feature selection techniques and boosting algorithms to enhance fetal health classification using CTG data. By employing RFE and evaluating models like XGBoost, LightGBM, AdaBoost, and CATBoost, the study finds that Random Forest with XGBoost achieves the highest accuracy (95%) and AUC (96%), demonstrating the effectiveness of ensemble learning. It also highlights the importance of feature selection in optimizing model performance and addresses class imbalance challenges using SMOTE. This research is highly relevant to our study, as it provides a benchmark for comparing SVM and ANN models with boosting techniques. Incorporating RFE for feature selection and evaluating cost-sensitive learning as an alternative to SMOTE can improve fetal health classification accuracy and clinical applicability, ensuring a more robust and interpretable predictive model.

[3] The study by Rawat and Mishra (2024) provides a comprehensive review of class imbalance handling techniques, which is crucial for fetal health classification where the number of Pathological cases is significantly lower than Normal cases. The paper explores data-level (SMOTE, undersampling), algorithm-level (cost-sensitive learning), and hybrid approaches, highlighting their impact on classification performance. Notably, cost-sensitive learning, which adjusts model loss functions instead of modifying the dataset, is presented as a viable alternative to SMOTE, making it particularly relevant for SVM and ANN models in our study. The research also discusses evaluation metrics such as AUC-PR, F1-score, and MCC, which are essential for assessing imbalanced classification performance. By incorporating insights from this study, our project aims to compare SMOTE vs. cost-sensitive learning to determine the most effective strategy for improving fetal health classification accuracy and fairness in machine learning models.

3 Methodology

This section outlines the complete process followed in developing, training, and evaluating machine learning models for fetal health classification. The pipeline comprises several key stages, including data preprocessing, feature selection, class imbalance handling, model implementation, evaluation, and interpretability analysis.

3.1 Dataset Overview

We use the *Fetal Health Classification Dataset* from the UCI Machine Learning Repository, consisting of 2,126 samples and 21 numerical features derived from CTG recordings. The target variable, `fetal_health`, is a multi-class label:

Label	Class
1	Normal
2	Suspect
3	Pathological

Table 1: Class distribution in target variable

Each feature represents statistical or physiological metrics related to fetal heart rate or uterine activity, such as accelerations, histogram measures, and deceleration patterns.

3.2 Data Preprocessing

3.2.1 Duplicate Removal

Duplicate records were removed using `pandas.DataFrame.drop_duplicates()`, ensuring no data leakage or bias from repeated observations.

3.2.2 Feature Scaling

All features were scaled using `StandardScaler` to normalize the range and center each feature around zero mean and unit variance.

```
from sklearn.preprocessing import
    StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

3.2.3 Train-Test Split

The dataset was split into 80% training and 20% testing sets using stratified sampling to preserve class distribution.

3.3 Feature Selection

To reduce overfitting and improve computational efficiency, feature selection was conducted using the ANOVA F-test via `SelectKBest`. The top 10 features based on their F-scores were retained:

Table 2: Top 10 features selected using ANOVA F-test

Selected Features
histogram_mean
histogram_mode
mean_value_of_short_term_variability
histogram_median
percentage_of_time_with_abnormal_long_term_variability
mean_value_of_long_term_variability
histogram_variance
abnormal_short_term_variability
accelerations
prolongued_decelerations

3.4 Handling Class Imbalance

3.4.1 Cost-Sensitive Learning

Implemented by assigning class weights inversely proportional to class frequencies using `compute_class_weight`. This was applied to both SVM and ANN models.

3.4.2 SMOTE (Synthetic Minority Oversampling Technique)

SMOTE oversamples the minority classes by creating synthetic examples:

```
from imblearn.over_sampling import SMOTE
X_sm, y_sm = SMOTE(random_state=42).
    fit_resample(X_train, y_train)
```

3.5 Model Implementation

3.5.1 Support Vector Machine (SVM)

SVM was implemented with an RBF kernel and class weighting. Hyperparameters were tuned using `GridSearchCV`:

Parameter	Values Explored
C	0.1, 1, 10
gamma	'scale', 0.1, 1
kernel	'rbf'

Table 3: Grid search parameters for SVM

3.5.2 Artificial Neural Network (ANN)

- Input Layer: 64 neurons, ReLU, Dropout(0.4)
- Hidden Layer: 32 neurons, ReLU, Dropout(0.3)
- Output Layer: 3 neurons, Softmax

Configuration: Adam optimizer, categorical cross-entropy loss, early stopping for overfitting prevention.

```
model = Sequential([
   Dense(64, activation='relu', input_shape=(
       X_train.shape[1],)),
   Dropout(0.4),
   Dense(32, activation='relu'),
   Dropout(0.3),
   Dense(3, activation='softmax')
])
```

3.5.3 ANN with Optuna Tuning

`Optuna` was used to optimize dropout rate, learning rate, and batch size.

Hyperparameter	Search Space
Dropout Rate	0.2 – 0.5
Learning Rate	1e-4 – 1e-2 (log scale)
Batch Size	{16, 32, 64}

Table 4: Optuna hyperparameter search space

3.5.4 XGBoost Classifier

Implemented with objective `multi:softprob` and metric `mlogloss`. Feature importance was analyzed using SHAP.

3.6 Interpretability with SHAP

To ensure interpretability, SHAP (SHapley Additive exPlanations) was used on the XGBoost model. This allowed visualization of both global feature importance and instance-level decisions.

- **SHAP Summary Plot:** Highlights average feature impact.
- **SHAP Bar Plot:** Ranks features by mean absolute SHAP value.
- **SHAP Force Plot (optional):** Explains individual predictions.

3.7 Data Visualization

To explore class separability, Principal Component Analysis (PCA) was applied to the feature space and visualized in 2D. Each point was colored by class label, providing insights into how well different classes can be distinguished visually.

4 Experiments and Results

To assess the effectiveness of different machine learning algorithms in fetal health classification, we conducted a series of experiments using Support Vector Machines (SVM), Artificial Neural Networks (ANN), and XGBoost. We evaluated the impact of **class imbalance strategies** (Cost-Sensitive Learning and SMOTE), as well as **hyperparameter tuning using Optuna**, and analyzed the results through various metrics and visualizations.

4.1 Evaluation Metrics

The models were evaluated using the following performance measures:

Table 5: Model evaluation metrics

Metric	Description
Accuracy	Overall prediction correctness
Precision	Class-wise positive predictive value
Recall	Class-wise sensitivity
F1-score	Harmonic mean of precision and recall
Confusion Matrix	True vs Predicted classes
ROC-AUC	Area under ROC curve for all classes

4.2 Results Summary

Table 6 give a overview of results. It clearly shows the different models implemented and the results obtained accordingly.

4.3 Support Vector Machine (SVM)

The SVM model with RBF kernel and cost-sensitive learning achieved strong overall performance with **90% accuracy** and a **weighted F1-score of 0.90**. It performed particularly well in classifying the "Normal" and "Pathological" categories. However, like most models, it had reduced precision on the "Suspect" class.

After applying **SMOTE**, the class recall for minority classes improved slightly, but overall performance remained comparable (accuracy dropped marginally to 89.1%).

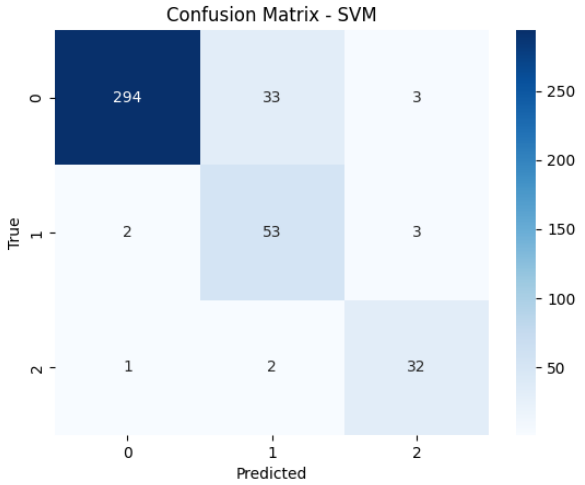

Figure 1: Confusion Matrix – SVM (Class Weighted)

4.4 Artificial Neural Network (ANN)

The initial ANN with dropout layers and class weights achieved 87% accuracy. However, it struggled with the minority "Suspect" class, showing a significant drop in recall despite decent overall F1-scores.

Applying **SMOTE** to the ANN significantly improved performance on underrepresented classes, achieving **90.3% accuracy** and **0.91 weighted F1-score**.

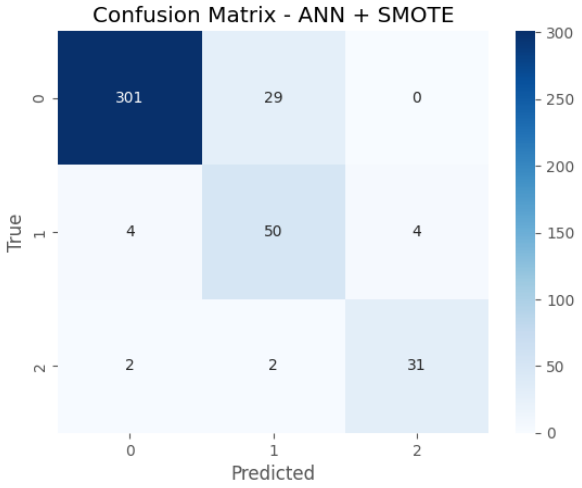

Figure 2: Confusion Matrix – ANN + SMOTE

We further used **Optuna** to optimize dropout rate, learning rate, and batch size. The best trial produced **89.5% accuracy** and **0.89 F1-score**, outperforming the untuned version.

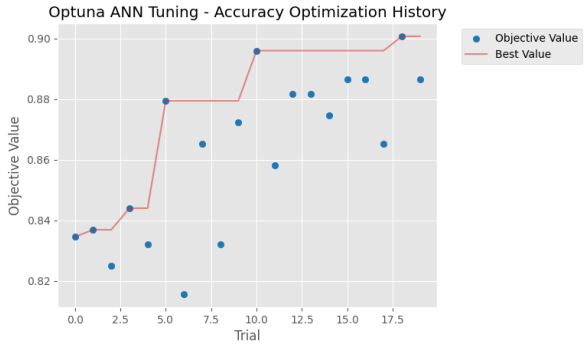

Figure 3: Optuna Hyperparameter Tuning History (ANN)

4.5 XGBoost Classifier

XGBoost delivered the best results across all models with **97% accuracy** and **0.97 weighted F1-score**, even without applying SMOTE or class weights. This performance highlights the strength of tree-based ensembles in handling minor class imbalances and learning complex nonlinear patterns.

Table 6: Performance comparison of all models on test set

Model	Accuracy	Weighted F1-score	Notes
SVM (class-weight)	90.0%	0.90	Balanced class weights
SVM + SMOTE	89.1%	0.90	SMOTE applied to training set
ANN (class-weight)	87.0%	0.84	No oversampling
ANN + SMOTE	90.3%	0.91	Improved recall on minority classes
ANN + Optuna (class-weight)	89.5%	0.89	Optimized dropout, LR, batch size
XGBoost	**97.0%**	**0.97**	No resampling; best performance overall

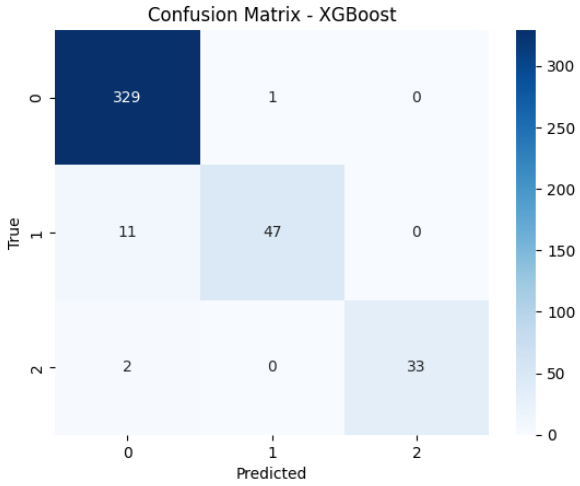

Figure 4: Confusion Matrix – XGBoost

4.6 Feature Importance with SHAP

To interpret the XGBoost model, we used SHAP (SHapley Additive Explanations), which revealed the most influential features in predicting fetal health:

- `histogram_mean`, `mean_value_of_short_term_variability`, and `abnormal_short_term_variability` were consistently top contributors.

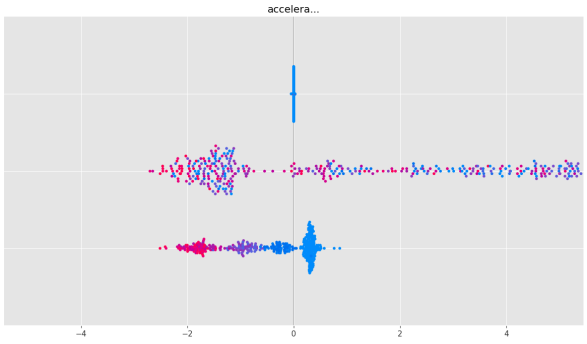

Figure 5: SHAP Summary Plot – Feature Importance (XGBoost)

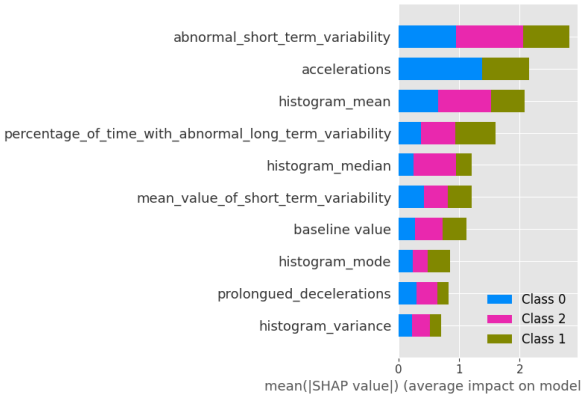

Figure 6: SHAP Bar Plot – Mean Absolute SHAP Value

4.7 PCA Visualization

To assess the natural separability of the classes in the feature space, we applied **Principal Component Analysis (PCA)** on the scaled dataset. The scatterplot in 2D shows that while there is visible separation between "Normal" and "Pathological" classes, the "Suspect" class overlaps significantly with both.

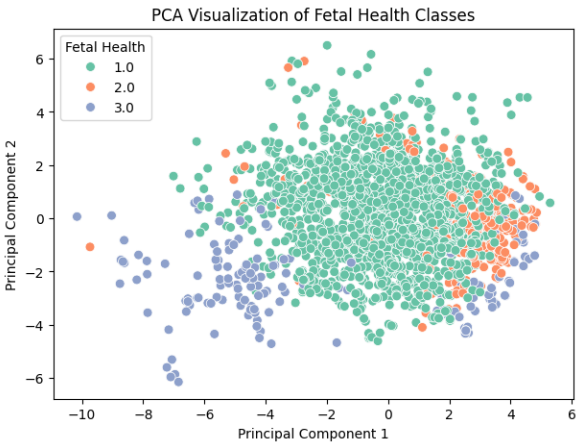

Figure 7: PCA 2D Projection of Fetal Health Classes

4.8 Model Comparison (F1-score)

A visual summary of the weighted F1-scores across models is shown below. XGBoost is the clear winner, followed by ANN + SMOTE and SVM.

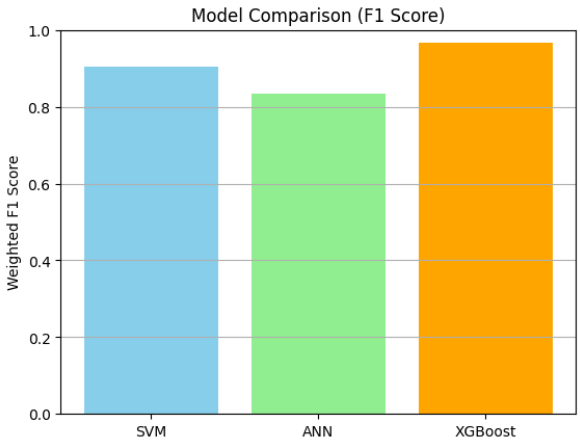

Figure 8: Bar Chart Comparing Weighted F1-Scores Across Models

5 Discussion

This section provides an in-depth analysis of the experimental findings, focusing on model performance, the impact of imbalance handling strategies, interpretability, and the broader implications of this study for clinical and academic use.

5.1 Comparison of Model Performance

Among the six models evaluated, **XGBoost** outperformed all others, achieving a **97% accuracy** and **0.97 weighted F1-score** on the test set. Its strong performance, even without class balancing methods like SMOTE or cost-sensitive learning, highlights the inherent robustness of ensemble boosting techniques. XGBoost was not only more accurate but also exhibited **high precision and recall across all three fetal health classes**, demonstrating its ability to handle non-linearity and class imbalance through internal regularization and tree-based learning.

The **Artificial Neural Network (ANN)** models showed significant variation in performance depending on the handling of class imbalance. The baseline ANN with class weights alone achieved only **87% accuracy**, struggling to correctly classify minority class instances, especially the "Suspect" category. However, after applying **SMOTE**, the ANN achieved a **notable performance boost**—up to **90.3% accuracy** and **0.91 weighted F1-score**—indicating that oversampling significantly improves the network's ability to generalize to underrepresented classes. This suggests that **ANNs are sensitive to class imbalance** and benefit greatly from balanced input distributions.

Support Vector Machines (SVM), on the other hand, showed **consistently strong but slightly lower performance** than XGBoost. Both the cost-sensitive and SMOTE-based SVM implementations yielded **around 90% accuracy**, with minimal change in weighted F1-scores. While SVMs effectively handled the imbalanced data via class weighting, they lacked the flexibility of deep networks or the ensemble strength of boosting algorithms, especially in capturing complex feature interactions.

5.2 Impact of Imbalance Handling

The results indicate that **handling class imbalance is crucial**, particularly for models like ANN that rely on gradient-based learning. Both **SMOTE** and **cost-sensitive learning** improved performance across the board, but **SMOTE was especially effective with ANN**, boosting minority class recall and overall F1-score.

For SVMs, **class weighting proved sufficient**, as there was minimal performance difference between the class-weighted and SMOTE-enhanced versions. XGBoost required no resampling at all, likely due to its internal boosting mechanism, which iteratively focuses on misclassified (often minority) samples.

These observations suggest that the **choice of imbalance handling technique should be model-specific**, with SMOTE benefitting neural models and class weighting being more appropriate for kernel methods.

5.3 Feature Interpretability

Interpretability is vital for clinical applications, where models must not only perform well but also provide **transparent justifications** for their predictions. Using **SHAP (SHapley Additive Explanations)**, we extracted both global and local feature importance scores from the XGBoost model.

SHAP revealed that features like `histogram_mean`, `mean_value_of_short_term_variability`, and `abnormal_short_term_variability` were the most influential in determining fetal health status. These findings are **consistent with clinical literature**, which emphasizes the importance of heart rate variability and abnormal deceleration patterns as key indicators of fetal distress.

Such insights validate that our machine learning models are learning patterns that **align with physiological understanding**, enhancing their credibility and applicability in real-world medical decision support systems.

5.4 Practical and Academic Significance

From a practical standpoint, this study demonstrates that **machine learning can effectively automate the classification of fetal health**, providing rapid, consistent, and high-accuracy assessments that can support clinicians in obstetric settings. Models like XGBoost can be integrated into fetal monitoring systems to flag

high-risk cases in real-time, potentially improving outcomes through early intervention.

Academically, this work provides a comparative evaluation of multiple learning paradigms on the same dataset, offering insights into the **trade-offs between interpretability, accuracy, and imbalance resilience**. Additionally, it illustrates the benefit of combining classical ML approaches with modern techniques like SHAP and Optuna to produce both **high-performing and explainable** models.

6 Conclusion and Future Work

This study explored the application of machine learning algorithms for the classification of fetal health based on cardiotocography (CTG) data. The primary goal was to evaluate the effectiveness of various supervised learning models—Support Vector Machines (SVM), Artificial Neural Networks (ANN), and XGBoost—in accurately categorizing fetal conditions into Normal, Suspect, and Pathological classes. The investigation further examined the impact of class imbalance correction techniques such as SMOTE and cost-sensitive learning, and employed interpretability tools like SHAP to make the models transparent and clinically relevant.

Among the models evaluated, **XGBoost emerged as the best-performing classifier**, achieving **97% accuracy** and a **weighted F1-score of 0.97** on the test set. Its performance exceeded that of SVM and ANN models even without the need for oversampling or class-weight adjustments, demonstrating its robustness to class imbalance and its capacity to model complex feature interactions effectively. ANN models, particularly when augmented with **SMOTE** or **hyperparameter tuning via Optuna**, also showed strong results, underscoring the importance of proper configuration and training strategies for deep learning models. SVM, while slightly less performant, remained consistent and interpretable with balanced class weights.

Additionally, the use of **SHAP for feature importance** proved critical in aligning model predictions with clinical reasoning. Features such as `histogram_mean`, `mean_value_of_short_term_variability`, and `abnormal_short_term_variability` were identified as the most influential in determining fetal health, aligning well with physiological understanding and previous research.

This work provides strong evidence that **machine learning can play a vital role in enhancing fetal health diagnostics**, particularly when used as a decision-support tool alongside traditional clinical assessments. Integrating such models into real-time fetal monitoring systems could potentially reduce misdiagnoses and allow for earlier interventions, improving maternal and neonatal outcomes.

6.1 Future Work

While the current results are promising, several areas for future exploration remain:

- **Dataset Expansion:** The current dataset is limited in size and scope. Future work should include larger and more diverse datasets, ideally with real-time CTG signals from various populations and geographies.
- **Temporal Modeling:** The existing models treat features as static. Given that CTG data is inherently sequential, applying **time-series models such as LSTM (Long Short-Term Memory)** or **GRU (Gated Recurrent Unit)** networks could capture temporal dependencies and trends in fetal heart rate more effectively.
- **Model Deployment:** Building a lightweight and efficient version of the final model for real-time use in mobile or embedded devices would be valuable, particularly for deployment in low-resource clinical settings.
- **Explainability for Deep Models:** Although SHAP was applied to XGBoost, future work can explore explainability techniques for ANN and LSTM models (e.g., Integrated Gradients or DeepSHAP), improving clinical trust in black-box models.
- **Cross-Validation Across Hospitals:** External validation using CTG data from multiple sources or hospitals would ensure the generalizability of the models and reduce the risk of dataset bias.

By addressing these areas, future research can advance toward building robust, interpretable, and scalable machine learning systems that significantly improve fetal monitoring and prenatal care.

7 References

References

[1] Yanjun Deng, Yefei Zhang, Zhixin Zhou, Xianfei Zhang, Pengfei Jiao, and Zhidong Zhao. A lightweight fetal distress-assisted diagnosis model based on a cross-channel interactive attention mechanism. *Frontiers in Physiology*, 14:1090937, March 2023.

[2] Sujith K. Mandala. Unveiling the unborn: Advancing fetal health classification through machine learning. *Artificial Intelligence in Health*, 1:2121, 2024.

[3] Satyendra Rawat and Amit Mishra. Review of methods for handling class-imbalanced in classification problems, 11 2022.

[4] Neny Sulistianingsih and Galih Martono. Enhancing predictive models: An in-depth analysis of feature selection techniques coupled with boosting algorithms. *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, 23:353–364, 03 2024.

8 Glossary

- AUC - Area Under the Curve
- AUC-PR - Area Under the Precision-Recall Curve
- ANN - Artificial Neural Network
- CTG - Cardiotocography
- CSL - Cost-Sensitive Learning
- FHR - Fetal Heart Rate
- MCC - Matthews Correlation Coefficient
- MBP - Maternal Blood Pressure
- RF - Random Forest
- RFE - Recursive Feature Elimination
- SVM - Support Vector Machine
- SMOTE - Synthetic Minority Over-sampling Technique
- UC - Uterine Contractions
- WPD - Wavelet Packet Decomposition